

FIG 1

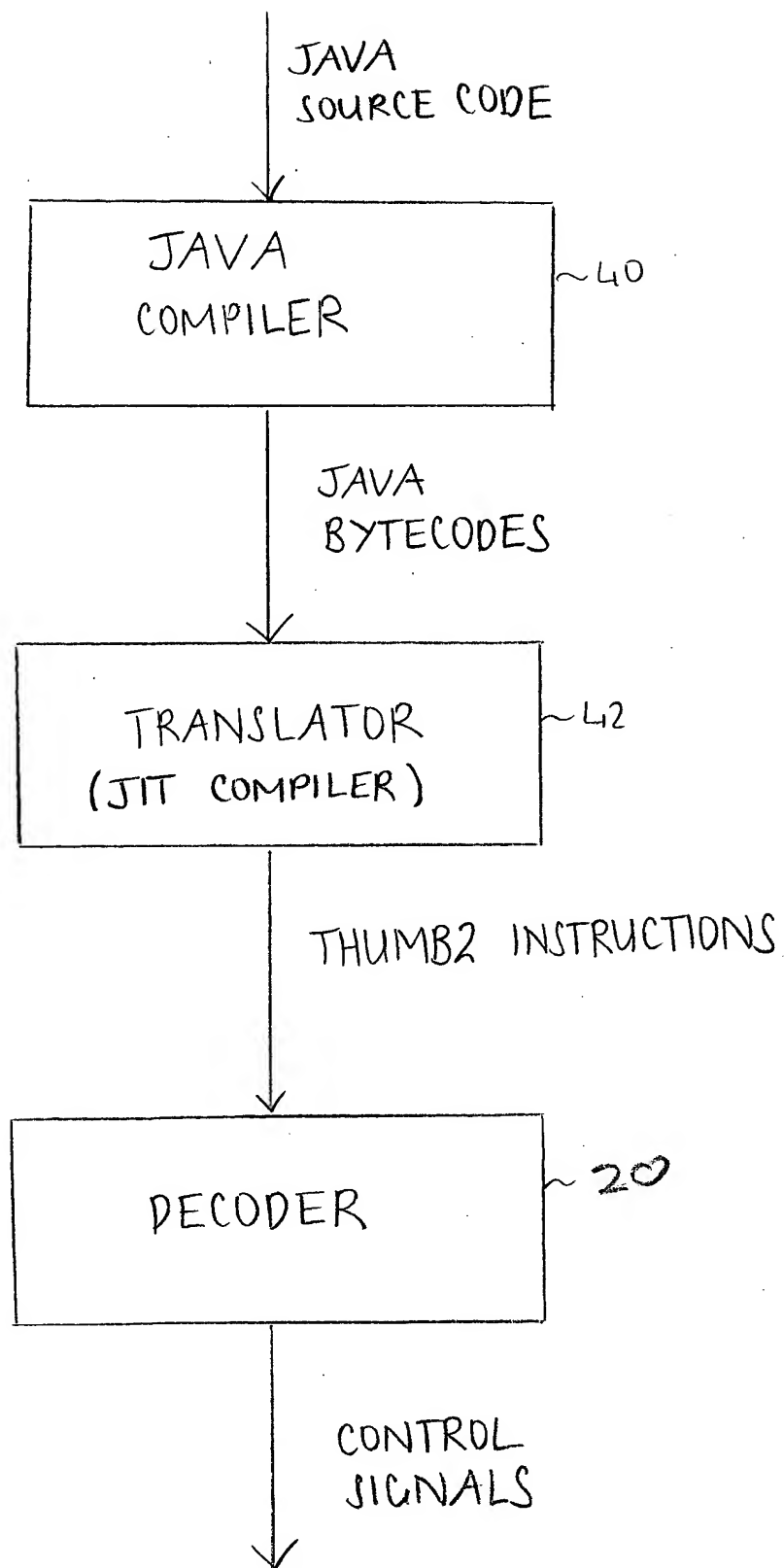


FIG 2

BYTECODE THUMB2 INSTRNS

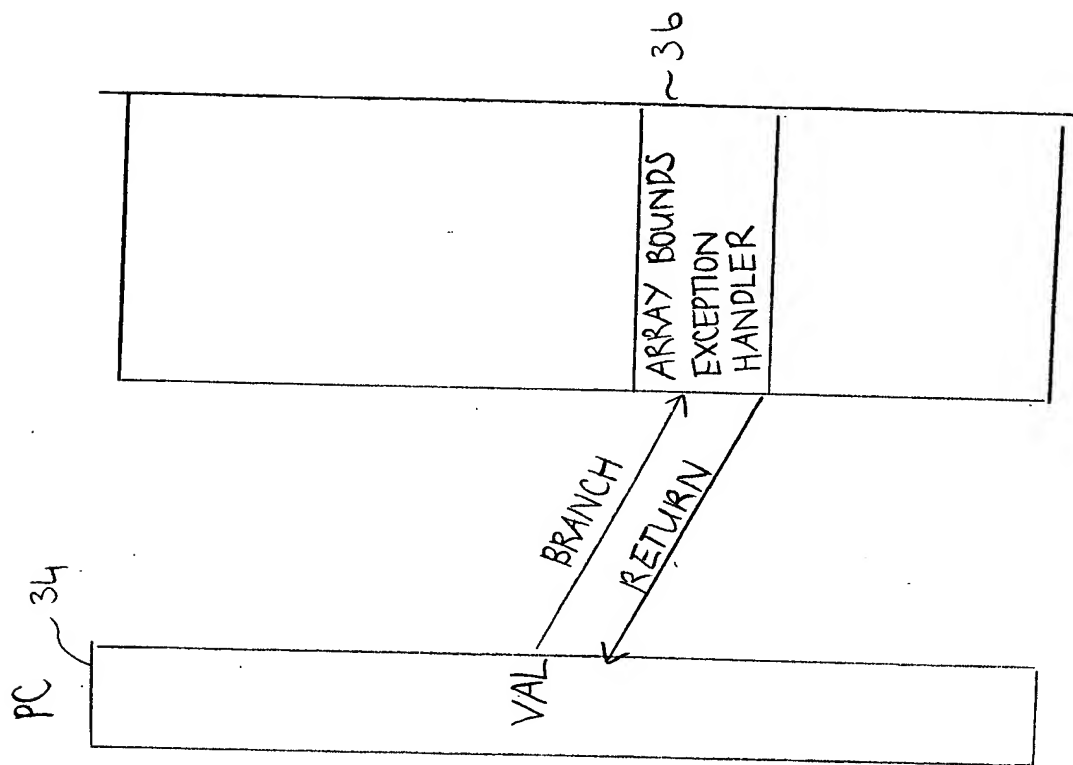
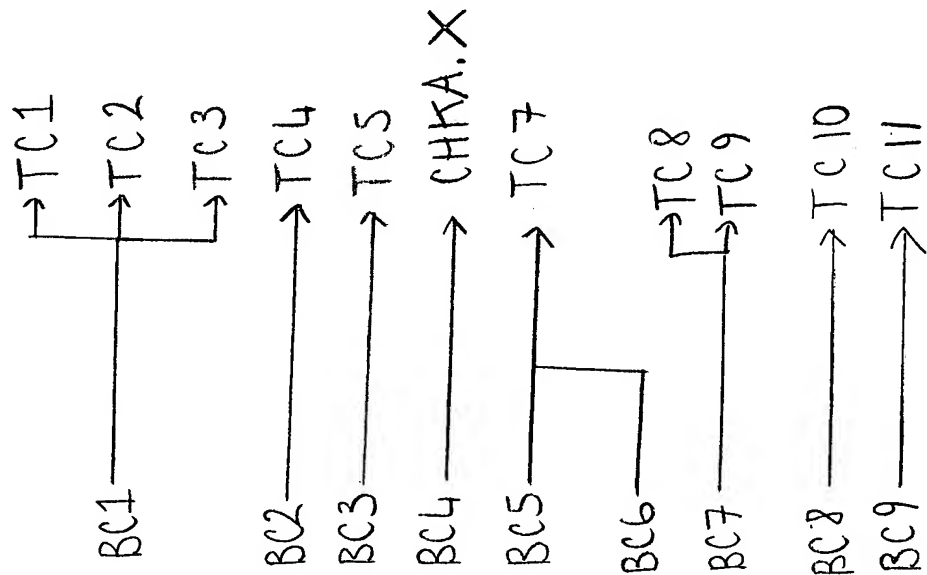


FIG 3

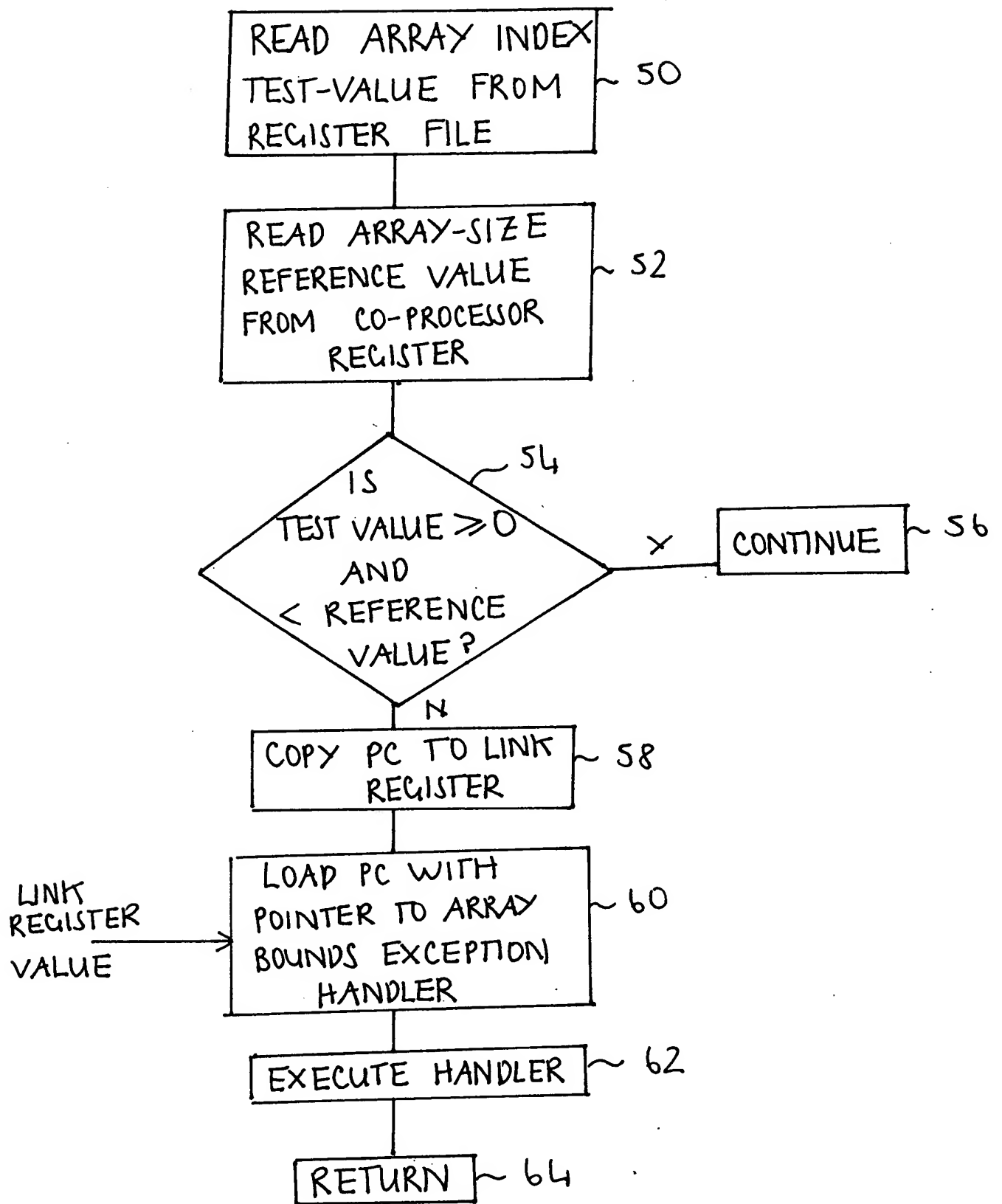


FIG 4

Instruction	CHKA.X Rn,Rm (16-bit)
Encoding	<div> <div>15 14 13 12 11 10 9 8 7 6 5 3 2 0</div> <div> <div>&lt; op code &gt;</div> <div> <div>H1</div> <div>H2</div> <div>Rm</div> <div>Rn</div> </div> </div> </div>
Thumb-2 Equivalent	<div>CMP Rn,Rm</div> <div>MOVLS lr,pc</div> <div>ADDLS pc,HandlerBase,#-8</div>
Definition	<div>IF (unsigned)Rm &gt;= (unsigned)Rn</div> <div>lr = pc</div> <div>pc = HandlerBase,#-8 ; IndexException</div>
Encoding space	<div>2^8</div> <div>8 bits</div>
Note	This is based upon the CMP(3) 16-bit Thumb-2 instruction that can use high registers
Note	H1 contains the most significant bit for Rn, H2 the most significant bit for Rm
Note	The LS case should almost never occur, so can be treated as exceptional behaviour
Note	This instruction does not set condition flags
Note	This comparison is UNSIGNED
Note	Return stack prediction will not be required when the MOV lr,pc step is executed.

FIG 5